1. INTRODUCTION.....5

About This Manual.....5
General Description.....5
Basic Technical Specification.....6
Dimension.....6
Interface.....6
Socket.....6
Optional adapter.....6
Power Supply.....7
System Requirement.....7
ChipMax Package Contains.....7

2. FAMILIES OF DEVICES.....7

EPROM.....7
EEPROM.....8
FLASH MEMORY.....8
Non-Volatile (NV) RAM and SRAM.....9
Serial EEPROM.....10
Non-Typical Devices.....10
8-bit 1-Megabits.....10
16-bit 1-Megabits.....11
Erasing an EPROM.....11
PLD.....11
PLD Features.....11
Microcontroller.....12

3. TERMS AND SYMBOLS USED IN THE GUIDE.....13

Safety Note Conventions.....13
Other terms and definitions are as follows.....13
Choosing The Right Adapter.....16
Different Device Packages.....16
Different Programming Adapters.....17

4. GETTING STARTED / INSTALLATION.....17

Installation Requirements.....17
Hardware Installation.....18
To install the software from the Internet download option.....18
To Install the software from a diskette drive.....18
To Start the windows software.....18
Trouble Shooting In Installation.....18

5. QUICK START EXAMPLES.....19

PROGRAMMING AN EPROM.....19
DUPLICATING AN EPROM.....20

6. OPERATIONS.....21

BASIC MENU SCREEN INFORMATION.....21
Option Information.....21
Additional Option Information For Non PLD Devices.....21
System Information.....21
Counter.....22
FILE.....22
- Binary Format.....22
- Intel HEX Format.....23
- Motorola S HEX Format.....23
- TEKTRONIX HEX Format.....24
- ASCII HEX format.....25
- JEDEC Standard <PLD devices only>.....25

File / Load.....26
File / Save.....27
Load Project.....27
Save Project.....27
Buffer.....28
Buffer / Edit Buffer.....28
Find.....28
Find Next.....28
Fill.....28
Copy.....29
Buffer / Edit UES.....29
Device.....30
Select.....30
- Select / E(E)PROM, FLASH.....31
- Select / PLD.....31
- Select / Microcontroller.....31
- Select / PROM.....31

Select / Auto Select.....32
Device / Blank Check.....32
Device / Program.....33
Device / Read.....35

Device / Verify....36
Device / Data Compare....36
Device / Erase....36
Device / Security....36
Device / Encryption....37
Device / Option....38
Device/Option /Customer ID....38
Device/Option / Oscillator....38
Device/Option / WATCHDOG TIMER (WDT)....38
Device/Option / POWER-UP TIMER....38
Device/Option / Memory Protect....38
Device/Option / Data Protect....39
Device/Option / Reset Polarity....39
Device/Option / Drown Out....40
Device/Option / MCLR....40
Device/Option / Memory Parity....40
Device/Option / Low Voltage PGM....40
Device/Option / FLASH Write Enable....41
Device/Option / Background DBG....41
Device/Option / Brownout Voltage....41
Option Item/Read Current Configuration Fuses....41
Option Item/Program Current Configuration Fuses....41
Device / Auto....41
Config....42
Config Option....42
Config Option / Default Buffer Value....42
Config Option / Buffer Clear Before File Loading....42
Config Option / Blank Check Before Programming....43
Config Option / Verify After Reading....43
Config Option / verify after programming....43
Config Option / Byte order swapping....43
Config Option / Device Insert Test....44
Config Option / Auto Repeat Programming....45
(under development)....45
Config Option / Port....45
Config Option / Gang Split Select....46
Split....47
Config Option / Address....49
File Load....49
File Save....50

Config / Hardware test....50

7. TROUBLE SHOOTING & TECHNICAL SUPPORT....51

1. Registration....51
2. Software Updates....51
3. Testing the Hardware....51
4. Quick Self-Diagnostics....51
5. Calling Customer Support....52
6. Service Information....52
7. Limited One-Year Warranty....53
8. Useful Web Site Addresses/ Phone Numbers....54
9. Programming Adapter Manufacturers....55
10. EPROM Emulator Manufacturers....55

8. GLOSSARY....55

1. INTRODUCTION

This manual describes the operation of E.E. Tools' **ChipMax** software driven device programmer. The information contained in this manual has been reviewed for accuracy, clarity, and completeness.
Please report in writing any errors or suggestions to support@eetools.com

E.E. Tools Inc.
550 Weddell Drive Suite 5
Sunnyvale, CA 94089
USA
Tel: 408-734-8184 , Fax:408-734-8185
www.eetools.com
E.E. Tools reserves the right to use and distribute any information supplied without obligation.

About This Manual

The ChipMax User Guide explains how to install and run the programming software on your computer.

- Chapter 2 describes the most popular programming devices.
- Chapter 3 contains all terms and symbols used in the manual.
- Chapter 4 contains instructions for installing and running the ChipMax.
- Chapter 5 describes basic operating examples of the ChipMax.
- Chapter 6 is organized by main operating commands and gives detailed instructions on each command including macro files and batch mode operation.
- Chapter 7 provides troubleshooting information for identifying and solving problems with your ChipMax. It provides a detailed guide for *E.E. Tools*' technical support and return material procedures.

This Manual assumes that you have a working knowledge of your personal computer and its operating conventions.

General Description

ChipMax is a software driven device programmer that supports a wide variety of programmable devices including: EPROM,

EEPROM, Serial PROM, EPLD, PEEL, GAL, FPGA, and single chip Microcontroller.
ChipMax easily connects to the parallel printer port of any IBM PC, and can operate with a full spectrum of IBM compatibles: PC XT, AT, 386, 486, Pentium, PS/2, portable(laptop), NT/2000, and clone computers.

The great advantage of ChipMax is its programming speed and superior software. **ChipMax** is controlled via a host IBM PC computer. The operating software has a user-friendly interface that includes window pull-down menus, macro facility for batch file executions (DOS version), and virtual memory management to deal with very large files (DOS version).
The main capabilities of ChipMax with software installed on IBM PC are:

- Selection of a device type and a manufacturer of source or target device,
- Loading and saving data files in Binary, Intel HEX, Motorola S, Tektronix HEX, and JEDEC formats,
- Distribution of 32- and 16- bit data into 8- bit portions,
- Maintenance of file, device, and buffer offset addresses,
- Blank checking, reading, writing, and verification of devices,
- Security fuse blowing for PLD devices,
- Provides parallel test vector application,
- Auto Device Selection of E(E)PROM and Flash memory devices,

  programming options,
- Hardware self-test for ChipMax programmer.

Basic Technical Specification

Dimension

- **Module**: 7.25”(L) x 5.00”(W) x 1.25”(H) (Perfect size for briefcase)
- **Material**: Steel
- **Weight**: 2.5 lb.

Interface

- **ChipMax** uses a standard parallel port (LPT1, LPT2, LPT3)

Socket

- **ChipMax** has a Textool 40 pin ZIF (300/600 mil) standard.

Optional adapter

- **CM-4G**: 4-GANG adapter for programming E(E)PROMs up to 32-pins

NOTE: The ChipMax universal pin driver capability lets it program all 40-pin DIP devices without DIP-to-DIP adapters, including all single-chip Microcontroller. The market, however, offers many choices besides DIP devices, such as PLCCs, SOPs, TSOPs, QFPs, and SOICs. We have developed over 50 different converters to support these non-DIP device packages. The ChipMax universal pin driver not only eliminates the need of programming modules, but also significantly reduces the number of converters required, reducing your cost for future expansion.

Power Supply

- Safety Standards: UL, CSA, VDE
- EMI Standards: FCC, VDE
- **Input:** 120VAC @ 60 Hz / 230VAC @ 50 Hz
- **Output:** 9.0 VAC 1.0 AMP

System Requirement

- **Type:** IBM PC XT, AT, 386, 486, Pentium, PS/2, Portable (notebook) or compatibles.
- **A hard disk drive**: 9Mbyte to install ChipMaxw.exe software.
- **RAM size:** 512K
- **Host computer I/O :** Either one of standard parallel ports (LPT, LPT2, or LPT3).
- **Operating System:** PC-DOS or MS-DOS (v 2.1 or greater).

Window95/98/NT/2000

ChipMax Package Contains

- ChipMax Programmer with a 40 pin ZIF socket.
- 6 feet long parallel printer cable.
- 120 VAC or 230 VAC power adapter.
- Installation diskette and manual.
- Registration and warranty card.

NOTE: Be sure to complete and return the enclosed registration and warranty card so that we can continue to provide you with updated software, technical support, and new programmer developments.

2. FAMILIES OF DEVICES

The devices that are supported on the *E.E. Tools*' programmers are:

EPROM

Ultraviolet erasable ROM's and use cell-based technology where each bit has a cell which is either charged or not charged to determine a logic one or a logic zero.
The structure of an individual EPROM memory cell is that of a dual gate metaloxide semiconductor (MOS). Stacked above the source and drain structure are two insulated gates. The upper gate is connected to the address multiplexer. The lower gate is left floating (electrically insulated from the surrounding components). If the cell has been erased (e.g. as shipped from the IC vendor), application of a logic high (+5V with respect to the source) to the upper gate will cause the current flow from the drain to source. A cell is programmed by applying a programming voltage (+12.5V to +25V depending upon the particular device) to the upper gate. This causes electrons to jump the insulating barrier and collect on the lower gate. When the programming voltage is removed, the device geometry is such that electrons remain on the lower gate. The electric field generated on the upper gate of a programmed cell is blocked by the electrons present on the lower gate,

preventing the current from flowing from the source to the drain.

EEPROM

Are devices which are electrically erasable and usually programmable at five volts. They typically have a part number starting with 28XXX.
EEPROMS, usually referred to as E-squareds, are Electrically-Erasable Programmable Read-Only Memory devices. EEPROMS program byte-by-byte (as opposed to FLASH memory which must be erased before programming). No erasure is needed before programming any location. The EEPROM automatically erases each location before it is programmed. EEPROMS have a slow programming time, are less dense than EPROMS or FLASH, and are usually used in situations requiring small amounts of memory due to their high cost. Although EEPROMS have slow programming times, some EEPROMS allow programming and erasure to take place in N-byte pages (where n=16,32, 64, etc…), making programming and erasure much faster.
For a comparison of EEPROMS with EPROMS and FLASH, see the selection under FLASH MEMORY.

FLASH MEMORY

These devices are also electrically erasable, but you must place the chip into erase mode and then erase parts of it. They require special algorithms to access the internal registers to erase or program. They typically have part numbers that start with 28FXXX or 29FXX. FLASH memory devices are unlike normal EPROMS in that they may be erased electrically. This means that it is no longer necessary to place the device into an eraser, but to simply erase it while it is in the ZIF socket. Since FLASH memory may be reprogrammed and is far less expensive than EEPROMS, FLASH memory is a good choice for in-circuit designs. One-time programmable (OTP) EPROMs are inexpensive because there is no window on the device. Because they can only be written into once it makes software updates either impossible or very expensive. EPROMs may be re-programmed, but only after they are taken out of the circuit and erased, which means that they cannot be soldered directly into the circuit, making production costs higher. Also,

EPROMs are more expensive due to the expense of the quartz window on the device. In some cases, such as a 32-pin (or smaller) PLCC, the device is too small to permit a window.
EEPROMs allow re-programmability, and thus may be directly soldered into the circuit. However, EEPROMs are considerably more expensive than OTP EPROMs, EPROMs, and even FLASH devices. EEPROMS are also less dense than EPROMs and FLASH.
FLASH memory, however, is relatively inexpensive as compared to EEPROMS, and may be re-programmed. This allows the device to be soldered directly into the circuit, avoiding the cost of having to remove the device.
FLASH devices may not be re-programmed byte-by-byte like EEPROMs, but when used for sequential storage or updates, it usually doesn't matter.
FLASH memory may be reprogrammed in 3 different ways: by erasing and reprogramming the entire device, erasing and reprogramming an entire block, or a smaller block of the device. INTEL and TOSHIBA flash memory usually need to be erased and reprogrammed entirely and requires 12 volts to program/erase.
SEEQ flash memory allows 512-byte blocks to be erased and reprogrammed independently of the rest of the device. SEEQ flash memory also require 12 volts to program or erase.
ATMEL flash memory allows 64-byte blocks of data to be erased and reprogrammed independent of the rest of the device. ATMEL flash devices program/erase with 5 volts. Also, ATMEL devices do not require special programming and erasing algorithms.
To erase a FLASH device with the ChipMax software, the option Erase must appear in the Device menu. If this is the case, press E to erase the device. With FLASH devices, the entire device is erased. *Before erasing, remember that the entire device will be erased.*

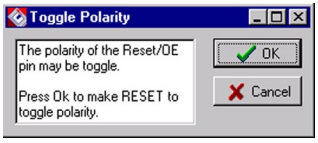
Non-Volatile (NV) RAM and SRAM

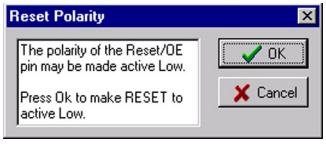
Are devices which are typically battery backed RAM and require 5 volts to program. They do not require erasure to program, and retain their memory after the power is turned off.

Serial EEPROM

These devices are electrically erasable, but they operate in a series rather than in parallell.

- **Xilinx 17xx family**
  From the Xilinx 17xx series, the RESET Polarity can be changed only on Xilinx 17xxD/L and 17128. On devices with EPROM portion already programmed or on new blank devices, RESET polarity is HIGH. The current status of the Reset pin polarity is determined and displayed on the screen after Reading the device. The polarity of the Reset pin can ONLY be changed from HIGH to LOW, but not vice versa. To change the polarity, select Option in the Device/Function menu and press **OK**, otherwise click on the **Cancel** button. To make certain that the RESET Polarity has been changed, read the device again. On other serial EEPROM devices (but NOT Xilinx 17xxD/L & 17128) the RESET polarity is always HIGH and it cannot be changed to LOW.





Non-Typical Devices

8-bit 1-Megabits

There are four types of 1 Megabits EPROMS. One set has the A16 and OE lines swapped. However, these devices will still program and verify like normal 1 Megabits. Once these devices are placed into the circuit, it will appear as if they have not been programmed correctly. This is not due to the ChipMax software or the ChipMax, but the difference between these 1 Megabits. When selecting a 1 Megabit, it is important to determine which one you have. Here is a list of 1 Megabits and their equivalents:

27010 (normal pin-out -- program as GENERIC or INTEL 27010):

Equivalents: INTEL 27010, HITACHI 27101, TOSHIBA 571000, NEC 271001, MITSUBISHI 27101

27301 (non-standard pin-out -- program as HITACHI 27301's): Equivalents: HITACHI 27301, NEC 271000, MITSUBISHI 27100, TOSHIBA 571001, INTEL 27C100

16-bit 1-Megabits

Any devices with the number 27210, 271024 and the MITSUBISHI 27102. 27011: The 27011 is a 28-pin 1-megabit device that is organized into 8 pages of 16k-bytes. NOTE: The 27513 is 4 pages of 16k-bytes.

Erasing an EPROM

An EPROM has a quartz window located on the chip just above the die. Erasing an EPROM is done by exposing the EPROM to high-frequency ultra-violet light waves. Erasing an EPROM usually takes from 15-20 minutes, but may be shorter or longer, depending on the device. Many manufacturers make EPROM erasers. If you wish to purchase an eraser, call E. E. TOOLS at 408-734-8184. When an EPROM is not being erased, the window may be covered with an opaque label. Sometimes (over a period of years) an EPROM will start to erase due to the rooms level of fluorescent light. Direct exposure to sunlight also has this effect, but happens much more rapidly.

PLD

A programmable logic device (PLD) consists of an array of logic gates and flip-flops that can be programmed to implement an almost unlimited number of logic designs. These are programmable logic arrays which can be EEPROM based, EPROM based, fused link, anti-fuse, or Flash-based technology. They are programmable by the user to implement logic circuits in order to reduce part count and turnaround time. PLDs are programmed according to a fuse map, which is typically contained in a JEDEC file.

PLD Features

Many different PLDs are available from the IC manufacturers. PLDs are fabricated using either bipolar or CMOS Processes. All PLDs are made up of combinations of AND gates, OR gates, inverters, and flip-flops.

- **PAL**: The PAL is a PLD with a fuse-programmable AND array. The PAL's AND gates connect to OR gates in a fixed pattern.
- **PROM**: For many years, the PROM was not classified as a PLD, even though most of the smaller PROMs (32 x 8 organization, for example) were being used as logic elements. The larger PROMs were still applied in bipolar microprocessor designs to store microcode instructions. The PROM has an architecture similar to the PAL, except that the PROM's AND array is fixed while its OR array is programmable.
- **FPLA**: The field-programmable logic array (FPLA) consists of a programmable AND array like the PAL, with a programmable OR array like the PROM. The FPLA is therefore a more general PLD because any product term may be connected to any output OR gate. Because the entire IC is programmable, the FPLA can implement some functions which a PAL or PROM may not be able to implement.
- **EPLD**: Several manufacturers produce PLDs which can be erased and reprogrammed like EPROMs. These ICs are called erasable programmable logic devices, or EPLDs. Internally, they have the same programmable AND-OR-register structures of the PAL and FPLA.

Microcontroller

These devices are CPU's with on-chip EPROM and RAM. They are typically 40 pins and are UV erasable. They have part numbers such as Intel's 8748,8749,8751,8752 etc. A micro-controller is generally a computer-on-a-chip with RAM, ROM, and I/O ports. Microcontrollers are usually used for specific purposes, such as keyboard decoders, printers, clocks, telephones, CD-players, or any other application that requires a small, on-board computer. Microcontrollers are used to take the place of in-circuit logic, as it can be less expensive and take less space. Also, since it is software driven, the device may be updated very easily. Micro-controllers have the ability to use internal as well as external RAM. Also, micro-controller data may be encrypted or otherwise secured to prevent copying of the data or program information. Micro-controllers also have

their own instruction set, usually very similar to familiar microprocessors (such as the 8080 or 8086). The INTEL MCS-51 family features up to 64k each of internal and external memory, 32 I/O lines, interrupts, timers, and bit-addressable RAM. It's instruction set contains 111 instructions. However, for specific purposes, limited versions of the 51 family are available. For instance, the Signetics 87c751/87c752 families do not allow external RAM to be used, and have limited I/O channels, etc. However, these devices still allow for data/program encryption and security levels. They are also less expensive than the MCS-51 micro-controllers.
See the help selection under MAIN-MENU COMMANDS for Encryption and Security-bit information.

NOTE: Programming Microchip PIC family
Microchip PIC series are different from other microcontrollers in that they have an EPROM area as well as a Configuration Fuse. The Configuration Fuse in the PIC family is used to setup different Oscillator types, to set Memory Code Protection and Watchdog timer, and etc. To program this fuse:

1. *Program the EPROM portion of the device*
2. *Click on* **Option**
3. *Make any changes if necessary*
4. *Click on the* **Program Configuration Fuses** *button to program the fuse information that you want to program*
5. *Click on the* **Read Current Configuration Fuses** *button to read back the current status of the fuse*
6. *Press the* **Close** *button*

NOTE: In order to obtain more information about programming the configuration fuse, please contact Microchip technology at 602-786-7200 or refer to their data book.

3. TERMS AND SYMBOLS USED IN THE GUIDE

Safety Note Conventions

■ ***NOTE*** *assists the user in performing a task. It makes the job more easily understood.*

■ ***CAUTION*** *alerts the user that unexpected results or damages to a device may occur if an instruction is not followed.*

Other terms and definitions are as follows

■ **Toolbar**: Clicking on a toolbar button manipulates operations or commands for ChipMax programmer software.

■ Bold/Italics – actions items/software functions, i.e. Edit Button, IC Test, or Change Algorithm.

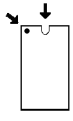
■ Device : The IC you are attempting to read, program, or verify.

■ Buffer : The work area in your computer memory to execute Read, Save, Program, and Verify. The Buffer size may be from 64K to 32 Megabytes.

NOTE: If the size of a device is bigger than the buffer size in your computer, ChipMax will use the hard disk space (swapping). For this reason, the ChipMax software can handle devices up to unlimited size of E(E)PROMs with your standard memory space (a minimum of 512KB RAM memory is required).

■ **Inserting a DIP Device**

Pin 1 on a DIP package is generally indicated by a notch on one end of the device. Pin 1 is also indicated by a stamped or recessed dot on one corner of the device. In the illustration below, the notch and the dot are highlighted by arrows. The dot is on the left and the notch is on the right.



- To Insert a DIP device into the ZIF socket:

  CAUTION: Devices are static sensitive. Operate your programmer at an antistatic workstation. To avoid electric shock and damage to the devices, use an antistatic wrist strap containing a 1 mega-ohm (minimum) to 10 mega-ohm (maximum) isolating resistor.

  1. Lift the socket lever to the open position.
  2. With the notched end facing the top of the socket, place the device in the socket so that the bottom of the device is aligned with the bottom of the socket (bottom justified).
  3. Lower the socket lever to lock the device into the socket.
  4. When you are done, select the device you just inserted.

- **Inserting a PLCC Device**

  *NOTE: In order to use **PLCC** devices, your programmer must have a proper adapter for the device. Consult your programmer's adapter list for the non-standard device.*

Pin 1 on a **PLCC** package is generally indicated by a notch on one corner of the device. Pin 1 is also indicated by a stamped or recessed dot on one side of the device. In the illustration below, the notched corner and the dot are highlighted by arrows. The notch is on the left and the dot is on the right.

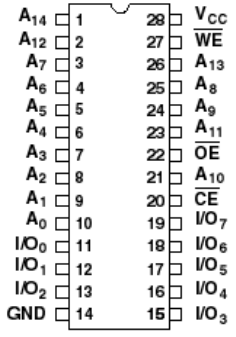
■To Insert a **PLCC** device into an adapter:

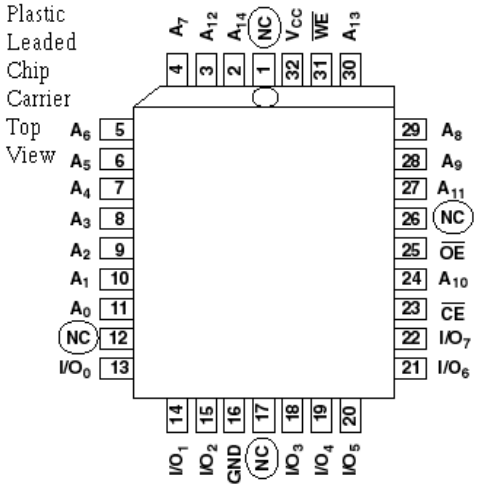
CAUTION: Devices are static sensitive. Operate your programmer at an antistatic workstation. To avoid electric shock and damage to the devices, use an

antistatic wrist strap containing a 1 mega-ohm (minimum) to 10 mega-ohm (maximum) isolating resistors.

1. Open the adapter 90 degrees, set its front edge under the two tabs at the front of the base opening, and lower its back edge into place.
2. Orient the device you want to use so that pin 1 is next to the retaining latch. Each Adapter has a small molded dot that represents pin 1 and a notched corner that can be used to align chamfered corners of devices.
3. Insert the device into the open adapter.
4. Close the Adapter, and press the retaining latch forward with your thumb until the latch snaps into place.
5. If you have not already, select the device you just inserted

Pin Connection Diagram Example between DIP and PLCC package



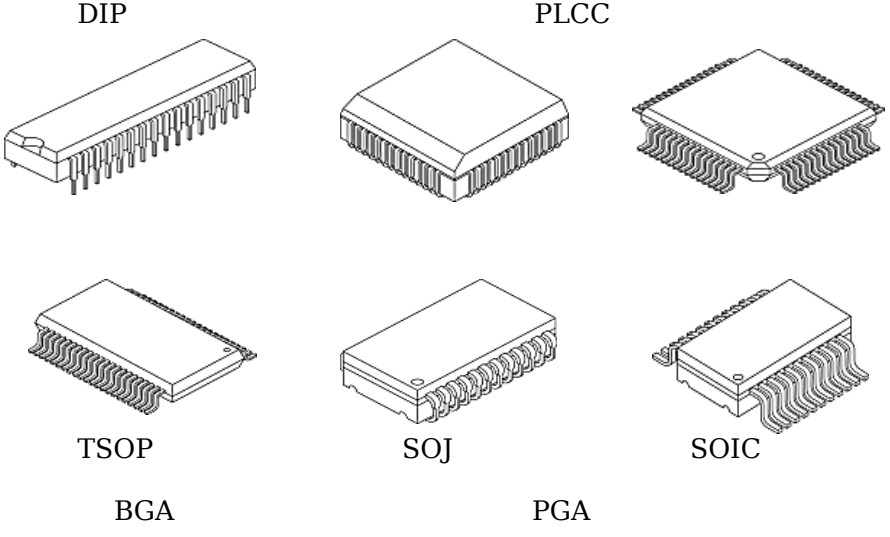


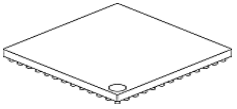
Choosing The Right Adapter

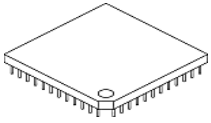
Most programming adapters are simple package converters. They allow QFP, SOIC, or PLCC devices to plug into the same device's DIP footprint. These adapters are available for memory, logic and microcontrollers. They can often be used with many devices from various manufacturers. For devices that cannot use a generic footprint we have offered adapters to work with specific programmers.
Here is what you need to know to select the appropriate adapter.

1) The part number and manufacturer of your device.
2) Your device package. (PLCC, DIP, QFP, SOIC, etc.) (Refer to the following package drawings)
3) Your device pin count.
4) In some cases you will need your device package dimensions for SOIC, SSOP, and TSOP chips.

Different Device Packages







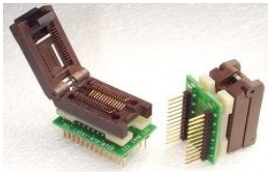
Different Programming Adapters

PLCC-TO-DIP

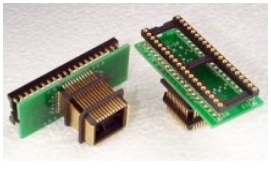
TSOP-TO-DIP

QFP-TO-DIP

SOIC-TO-PLCC



DIP-TO-PLCC



(for Emulator)

4. GETTING STARTED / INSTALLATION

Installation Requirements

ChipMax is designed to operate with any IBM-PC, XT, AT, 386, 486, Pentium, PS/2, Portable (notebook), compatibles running PC-DOS, MS-DOS 2.1 or greater, WIN NT, WIN2000. The computer requires a double-sided disk drive (1.44MB), but a hard disk drive is also recommended

Hardware Installation

The following section details the procedure for accomplishing the hardware installation procedure. ChipMax easily connects to any parallel printer port in your computer. There are three different addresses for the parallel port. When you select an address from LPT1, LPT2, LPT3, one of them should be valid without a communication error message. Turn the AC switch ON before running the ChipMax software. Make sure that you connect the printer (IEEE) cable between ChipMax and your available printer port and lock the shields in each side of the cable. Be sure that ChipMax recognizes your computer's parallel port address when you execute the ChipMax icon.

To install the software from the Internet download option

- Go to www.eetools.com and select software ***CHIPMAXW.EXE*** from the download section when you are ready to download.
- When prompted, choose the drive and directory where the self-extracting CHIPMAX.EXE file is to be saved in.
- Once the download is complete, double-click on the file name to install the software.

To Install the software from a diskette drive

- Place the diskette labeled ½ in the diskette drive.
- Choose SETUP.EXE from the list of files located on the diskette. The SETUP program will launch the installation procedure and it will ask you to insert the second diskette. Follow the same procedures as for the first.

To Start the windows software

- To run the windows software, select the ChipMaxw shortcut in the Windows Start Menu/Programs list.

Trouble Shooting In Installation

A communication error may occur on the screen if the hardware / software is not correctly installed.
Be Sure That The Following Steps Are Checked:

- Make sure that the **ChipMax** is connected to your PC printer port directly. It will not work with multiple port connector.
- Be sure your printer cable is firmly connected to your computer and **ChipMax** programmer.
- Plug in the AC power cord to your **ChipMax** and turn the switch in the back of the ChipMax on.

NOTE: The CHIPMAXW detects the printer port address when you install the new software. When you see "Cannot find the ChipMax programming module", go to CONFIG/PORT and select all three parallel port addresses. If you see the same error message continuously, you should contact technical support.

5. QUICK START EXAMPLES

If you are using the ChipMax for the first time, this section will help you to become familiar with the basic operating procedure. This section includes two examples of device programming with your ChipMax.

Programming an EPROM

We selected an AMD 27C010 EPROM to show you how to program an EPROM. The 27C010 EPROM needs to be erased (blank) before this procedure begins.

NOTE: EPROMs have a quartz window that can be erased by exposing the EPROMs to ULTRAVIOLET(UV) light. Erasing an EPROM usually takes 10-30 minutes.

1. Click on the ChipMaxw shortcut :

2. Check the optional configuration before programming begins.

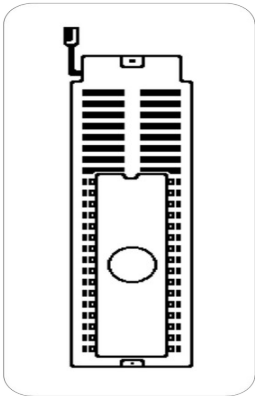
3. Click on the **Select** button. There are two different ways to select the target device from the menu: by choosing the device manufacturer and type using the arrow keys or you may type the manufacturer and the device names on NAME box. ChipMax will display the names of devices that have the best match to your input. After selecting the device, the detailed device information box is provided below the select menu screen.

4.Click on the **Load** to load a file from a floppy or hard disk into the buffer. Change your file directory by choosing a directory in LOOK IN box. Make sure that the file type is selected; "All Hex File" or "Binary file" is located in the File of type box.

5. Select the ***Device*** in order to begin programming.

6. Insert the 27C010 device into the ZIF socket. After inserting the part, make sure that the socket handle is down (close) to secure the chip.

See the illustration below:



7. Click on the highlighted cursor ***Blank Check***.

NOTE: If an EPROM is not erased completely, it will not pass the Blank Check. If an EPROM is damaged to

begin with, it may not pass the blank check, although it has been erased for a long time.

8. Click on the ***Program***.

CAUTION: Do not touch the device while the BUSY GREEN LED light is on (programming is in progress).

After programming a device, the part is automatically verified. The CheckSum is calculated and displayed in the OPTION info. In order to verify your work, read the programmed part again. If this CheckSum value matches to that of the Programming CheckSum, then the 27C010 is programmed successfully.

Duplicating an EPROM

The following is an instruction on duplicating a programmed device. In order to do so a source device and an **erased (blank)** target device are necessary.
Source device : Programmed AMD 27C256
Target Device : Erased or blank INTEL 27C256
1. Make sure the **ChipMax** main menu is displayed without any communication error (refer to programming section).
2. Place the AMD 27C256 device into the ZIF socket.
3. Select the manufacturer and part names from the **Select** menu.

4. Click on the **Read** button. In order to make sure the device is read properly, Click on the **Verify** button.
5. Remove the current chip from the socket and replace it with the erased or blank INTEL 27C256 device. Select the appropriate device from **Select** menu on screen.

NOTE: You do not need to change the device information if you use the exact same chip as the source device.

6. Click on the **Blank** button.
7. Click on the **Program** button. The part will be programmed and verified automatically. If no error messages appear during the Programming or Verification process, your duplicating work is done successfully. You have a duplicated INTEL 27C256 part from AMD 27C256 chip.

6. OPERATIONS

This section describes the operation of the software. The Main standard system-menu is divided into four display areas: Main operation menu screen, Option Information, System information, and counter.

Basic Menu Screen Information

Option Information

- **Gang Size** : Current socket size when ChipMax is used
- **Split** : Current world format for split programming
- **Enc Mode** : Enable or Disable Encryption mode for Microcontrollers
- **Base Port** : Current parallel port address
- **Check-Sum** : Check-Sum number of the data in current buffer
- **H/W Rev** : Hardware revision number for your programmer
- **S/W Rev** : Current **ChipMax** software revision number
- **Serial No** : Serial number of **ChipMax** hardware

Additional Option Information For Non PLD Devices

The following information presents programming information of the selected device.

- **Algo** : Programming Algorithm
- **Vccp** : Main Power Supply Voltage
- **Vpp** : Programming Power Supply Voltage
- **Tpwp** : Programming Pulse Width

System Information

- **Manufacturer** : Manufacturer name of the current device
- **Device Name** : The current device number
- **Device Size** : The size of device in HEX value (Ending Address – Starting Address + 1)
- **Free Disk** : Check the free disk space for a big size E(E)PROM programming.

- **Adapter Name** : Optional Adapter Name for Non-standard devices
- **Num. Of Pins** : Number of device pin
- **File** : Current working directory path and file name after loading a file

Counter

- **Devices/HR** : Displays the estimated number of devices that can be programmed per hour. This feature can only be used when choosing the *Program* or *Auto* selection under the Device button.
- **Success** : The number indicates the devices programmed successfully.
- **Failure** : This number indicates the number of device programming errors that occur during a programming cycle. These could be either ***Blank Checking***, ***Programming***, or ***Verification*** *error*.
- **Count** : The number indicates all devices executed successfully and unsuccessfully.

NOTE: This feature is useful for repeat programming on the same device. You can make an estimate time to perform the programming job and see the successful and failed devices after finishing the Program or Auto Repeat programming routine.

File

ChipMax uses three different file types: BINARY, ALL HEX, and .POF.

In the file types box, a file type can be selected and loaded to the buffer or saved onto a disk. The default file type is the Binary file. The All HEX files can be chosen by maneuvering the arrow button. All HEX files include INTEL HEX (MCS-80/86/386, MOTOROLA S (1-9), Tektronix HEX and ASCII HEX. OPF (Programmer Object File) is a binary file generated by Altera assembler (Quartus and MAX+PLUS II). This file should be loaded for Altera MAX or EPC family devices only.

■ Binary Format

Binary format does not specify the address or checksum of the file. The file contains the actual binary data. An example of this format is a DOS executable file with an .EXE or .COM extension. Binary format is generated for programmable memory devices. It is recommended to save your EPROM data as binary format in order to load the file as a standard file format later.

■ Intel HEX Format

Intel HEX format files are text files that include the file information in hexadecimal.

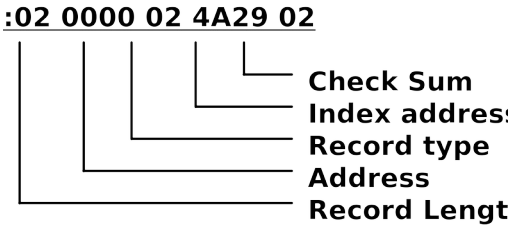
1	:	A record mark
2 – 3	Byte	Record length in 2 digit HEX, Max 20 (64 in ASCII)
4 – 7	Address	4 digit HEX Field. Most significant byte first
8 – 9	Byte	2 digit field record type : 01 End of file 02 Extended address
10 – N	Data	Data field in HEX digits
N+1 – N+2	Check-Sum	Two digit HEX Check-Sum character computed by two's complementing the sum of previous bytes except the ':'

INTEL HEX FILE EXAMPLE

:11000000044415441 4D414E20533320534552494 14C73

:00000001FF

The extended address record specifies the index address where data will be loaded into. The Extended Address will continue to offset data record address until a new Extended Address record is specified.



- The Address field is blank because this record is not data.
- The record length is '02' for index address (2 Bytes).

  NOTE: If the address for the data record is '2B56', the actual address will be 4A290 + 2B56 or 4CDE6(HEX).

■ Motorola S HEX Format

The Motorola S format file is an ASCII-HEX file.

Position (Byte)	Character	Remarks
1	S	Letter S indicates start of record
2	0, 1, 2, 3, or 9	A single character indicates the type of record. **9**: End-of-file **3**: 32-bit address data record **2**: 24-bit address data record **1**: 16-bit address data record **0**: Header
3 - 4	Byte	Byte COUNT in HEX (multiply by two for number of characters). This count includes the address, data, and CheckSum field.
5 - X	Bytes	Memory Address for the current record. **X** will be: **8** : 16-bit addressing for files less than 64K.

10: 24-bit addressing for files greater than 64K.

12: 32-bit addressing for files greater than 64K in length.

X+1 - N	Bytes	HEX Data (two per byte)
N+1 – N+2	Check-Sum	Two digit HEX Check-Sum character calculated by one's complement of DATA, ADDRESS and COUNT.

Motorola File Example

S1140000444154414D414E2053332053455249414C6F

S9030000FC

■ TEKTRONIX HEX Format

The Tektronix HEX format contains ASCII records, expressing bytes ASCII pairs.

Position	Character	Remarks
1	/	Slash character for start of line
2 - 5	2Bytes	Address. MSB first load
6 - 7	Byte	Number of data bytes (not checksums)
8 - 9	Byte	Check-Sum of ADDRESS and COUNT by character in HEX (not by byte)
10 - N	Data	Data bytes as ASCII pairs
N+1 - N+2	Byte	Check-Sum of Data by character (not as bytes)

Tek Hex Example

/0000110244415441 4D414E2053332053455249414C8F

/01000001

■ ASCII HEX format

This selection generates an ASCII coded HEX format for either 4-bit or 8-bit PROMs. Each record contains a four-digit HEX address (16-bit) followed by 16 data elements. A 16-bit checksum is at the end of the file.
When this format is selected, the device base address must be specified. This address represents the lowest address in the device. The file created contains an entry for each location in this device. ASCII HEX format can be created for programmable memory devices only.

■ JEDEC Standard <PLD devices only>

JEDEC (Joint Electronic Device Engineering Council) files are the standard method for describing PLD fuse patterns and test vectors. JEDEC files contain fuse data, test vectors, part numbers, and checksums. The checksum of the file allows you to verify that a given file is intact and has not been unintentionally modified. JEDEC files normally use the extension (last 3 letters) ".JED."
For more information on the JEDEC standard, contact:
Global Engineering Documents Inc. at (800) 854-7179
Electronic Industries Association at (202) 457-4900.

Following is an example of a JEDEC file:

```
<STX>File for PLD 15S8 Created on 11-SEP-96 5:08PM
2754 memory decode 345-432-123
Seung Park PK Logic corp.
QP20* QF448* QV8*
F0*X0*
L000011111011111111111111111111*
L002810111111111111111111111111*
L005611101111111111111111111111*
L011201011011011101111111111111*
L022401111011011101111111111111*
L033601010111011101111111111111*
V0001000000XXXNXXXHHHLXXN*
V0002010000XXXNXXXHHHLXXN*
V0003100000XXXNXXXHHHLXXN*
V0004110000XXXNXXXHHHLXXN*
V0005111000XXXNXXXHLHHXXN*
V0006111010XXXNXXXHLHHXXN*
```

V0007111100XXXNXXXHHLHXXN*
V0008111110XXXNXXXLHHLXXN*
C124E*<ETX>8646

STX The fuse map begins with an ASCII STX character (02 HEX)

Design Specification This item is user specified. While no format rules apply, certain information, such as user's name, company, design date, part designation, revision and device part number, should be entered. This field is illustrated by an asterisk (*).

QP Specifies the number of pins in the devices.

QF Specifies the number of JEDEC fuses in the devices.

L The fuse list fields contain the state of all fuse links in the devices. The starting fuse number follows the L specifying the field type. The fuse list that follows contains a zero (0) for each intact link and a one (1) for each blown link. An L field is generated for each product term in the device.

C The checksum field contains the 16-bit sum of the link stated in the 8-bit words.

ETX The fuse map ends with an ASCII ETX character (03 HEX).

Sum Check A 16-bit sum of the ASCII values of the characters from STX to ETX inclusive. The sum check follows the ETX.

NOTE: LOGIC Compilers For PLD Devices: Software is available to help the engineer develop designs using PLDs. Software tools called logic assemblers or compilers translate a design file written in high-level language into a fuse pattern stored in a JEDEC file. JEDEC files are produced by almost all PLD development software's and are accepted by the ChipMax programmer. There are many commercial software packages available to help you design using PLDs.

File / Load

Data can be loaded into the memory from a device or by opening a data file. Load fills your buffer memory with the data from storage for viewing or editing. This command loads the data from the selected file storage into the memory buffer. In order to the use "All HEX File" selection, the HEX file must be one of the file formats supported by the ChipMax, such as Intel HEX(MCS-80/86/386, MOTOROLA S(1-9), Tektronix HEX and ASCII HEX.
The default selection on File Load menu is in Binary Format. To select any of the HEX files mentioned above, choose "All HEX File" by pressing ⇓ button. When you have selected the desired file, press the OPEN button to load the file into the data buffer. If you are programming a PLD, you will want to load a JEDEC file. The procedure is identical to loading a data file, except that the files in the current directory will have the JED extension. If your selected device is an Altera MAX family, the file you should load is a POF extension. The ChipMax uses a RAM buffer to hold data. After loading a file into the buffer, you can edit the buffer data. If you load a JEDEC file, you may

use (the vector pattern edit) command to view or edit the fuse map and (test/vectors) for any test vectors that may have been in the JEDEC file.

File / Save

Save the current data in your memory buffer to a disk storage by using one of the current supported file formats.
Before saving a file, check the buffer and the file address ranges. The contents of the buffer through the specified range will be written into the new file, completely erasing any

existing file with the same name. Before saving to a disk, make sure that no file with the same name exists.

Load Project

A project file that saved by SAVE PROJECT menu is loaded. The project files use the extension (last 3 letters) ".prj."

Save Project

This feature allows you to create a job description such as "engineer name" and other useful information for records. It is very useful for future use when you set up all possible environments such as selecting a device, loading a file, and setting other configurations for programming jobs. A job description can be saved as a file name and the same project environment will be ready once you load the same project name.
File Name: A file name can be entered with the 3 letter extension ".prj."
Author: An engineer's name [whom creates this project].
Description: A job explanation that you memorize for your future usage. A device number, File name, and checksum number can be entered in the note pad. Other programming menu descriptions, such as configurations can be described.

Buffer

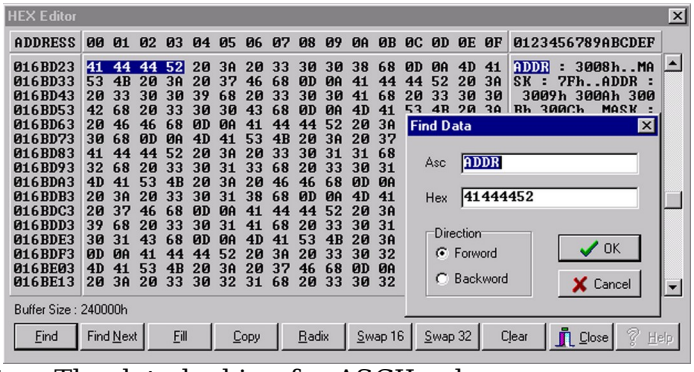
Buffer / Edit Buffer

This command allows the user to examine and modify the contents of the memory buffer. This section applies to a non-JEDEC file (PROM, EPROM, EEPROM, and Microcontroller) or to a memory chip. If a PLD is being loaded, see the (vector pattern edit) section. The data is presented in HEX and ASCII formats.

Find

This feature allows you to search the data (ASCII and HEX) in the current

buffer.



Sac: The data looking for ASCII value.
HEX: The data looking for HEX value.
Direction UP: The data searching from previous address than the current location.
Direction DOWN: The data searching from higher address than the current location.
If you would like to find more data, click on the **Find Next** button.

Find Next

Press the **Find Next** button to locate the rest of the data that you entered in the **FIND** box. The error "Search Pattern not Found" will be accursed when you press this button without entering data in the **FIND** text box.

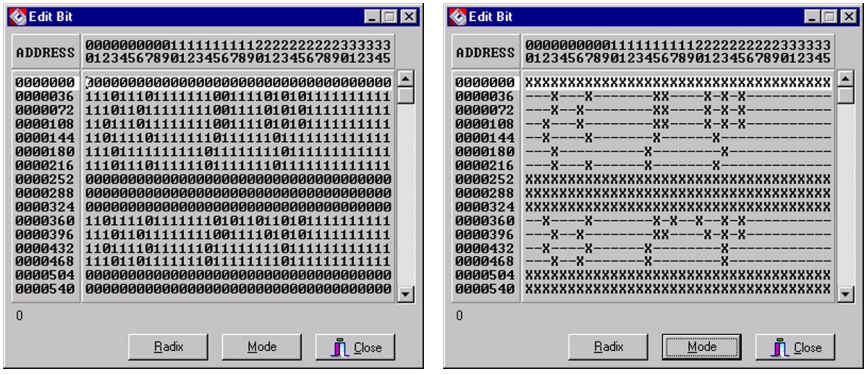
Fill

Buffer Start: Starting address for the data to be filled in buffer.
Buffer End: Ending address for the data to be filled in buffer.
Fill Data: Two digits of HEX value to be filled between Start and End buffer.

Copy

Buffer Start: Starting address for data to be copied.
Buffer End: Ending address for data to be copied.
Destination: Start address for data to be pasted.

Radix: Pressing this button will display the address value in decimals (using 0-9) or HEX (using 0-9,A-F). The information appears on ADDRESS column at the left of the buffer screen.



Swap16 This option applies to the current buffer data. Pressing this button allows you to swap 8bit data from ODD address to EVEN address.

Swap32 Pressing this button allows you to swap 16bit data from ODD address to EVEN address.

Clear Pressing this button allows you to fill the buffer with the data located in "Default Buffer Value" in Config Option Menu.

Close Press to exit the HEX Editor.

Buffer / Edit UES

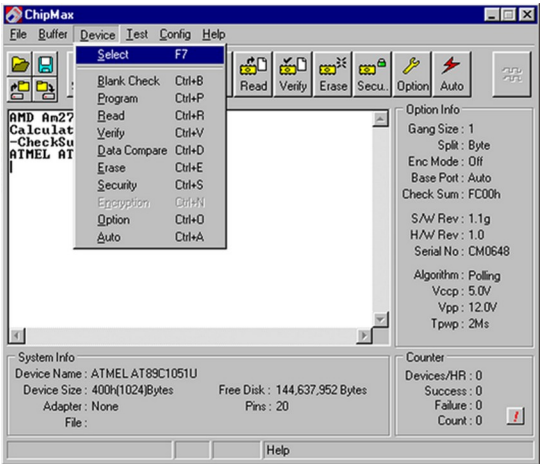
The UES Edit command creates or changes the User's Electronic Signature (UES) array in GAL device. Each GAL device contains an electronic signature word consisting of 64 bits of reprogrammable memory. The electronic signature word can be programmed to contain any identification information desired by the user. Some uses include pattern identification labels, version numbers, dates, inventory control information, etc. These features give the user the ability to view and edit the UES data before programming a GAL device. When the UES edit command is invoked, an editing data window appears. If the data fields are empty, you may create a new UES. You can enter the UES up to eight characters in the

HEX or ASCII data area. If you see any data from the current UES window, it means the UES has been created and that you can modify the data for a different reason. The UES data is not secured when you execute the Function / Security command.

Device

This section presents the main operation menu for the target device that is mounted on the ZIF socket. In order to process the following commands, make sure that the device is correctly inserted into the ZIF socket and the latch is down.

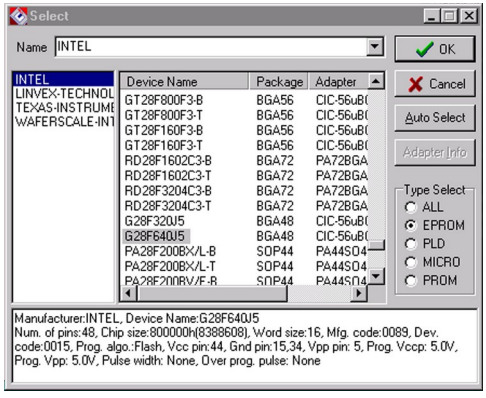
NOTE: The Device Information display area presents the device information of the selected device.



Select

During operation, the first step is usually to select a device. This Select command enables the user to define the manufacturer and the type of the device that will be used. After you select a device, you can insert a device into the programmer's device socket and conduct various device operations such as programming and verifying device data or reading data from the device. The Select command contains both manual and automatic methods for selecting a device. If your device is not identified by the Auto Device Select menu, you can select the device list displayed in the Manufacturer & Device list. Scroll through the manufacturers and device

numbers until you find the manufacturer and device you are looking for. You can use wildcards to help you "zoom" on the device you are looking for.



■ Select / E(E)PROM, FLASH

All EPROMs (27xxx), EEPROMs (28Cxxx, 29Cxxx), Serial E(E)PROMs (17xxx, 24xxx, 32xxx, 33xxx, 35xxx, 59xxx), and Flash EPROM (28Fxxx, 29Fxxx) of 24/28/32/40/42 and up to 40 pins (1 Mbit, 2Mbit, 4Mbit, 8Mbit,16Mbit, 32Mbit, and up).

■ Select / PLD

GAL16V8,GAL20V8,GAL22V10
PALCE16V8, PALCE20V8, ATF750

■ Select / Microcontroller

Intel 87xx, Phillips 87C75x,
Atmel AT89Cxx, 89Sxx,
Microchip PIC12/16/17

■ Select / PROM

Atmel 27HC641,
ICT27CXxxx, WSI 57Cxx.

Select / Auto Select

Auto Device Select
Warning :
Auto Device Select is an optional feature.
This option is useful for only 32-pin or less than 32-pin EPROM and Flash Memory.
Other types(GAL,EPLD,EEPROM,Microcontroller) of devices are not able to be recognized by this option and the devices may be defective when you use this feature in the programmer.
If you know the manufacturer and part number, select your part number manually.
OK
Cancel

Identify the device that is mounted on the ZIF socket. This feature can only be applied to Memory and some Microcontroller devices. Clicking the Auto Select button will enable the programmer to identify the ID on the device and will select the matching device in the library automatically.

> *NOTE: If you have a "Device not found" message, select the device manually. If you have old devices or defective devices, the **ChipMax** will not be able to recognize the ID code from your device.*

Device / Blank Check

The Blank Check function is used to verify whether or not a device is in an erased or unprogrammed state.
All EPROM (Erasable Programmable Read Only Memory) devices should be checked before programming. EEPROM (Electrical Erasable Programmable Read Only Memory) based parts do not need this command because EEPROM's are erased automatically before programming.
PLD based parts are checked by verifying all of the fuses that are intact. Any erased PLD's should pass this test.

> *NOTE: Erasing EPROMs. In order to clear data in an EPROM, the chip should be exposed to a short wave UV (Ultra violet) light. Most erasers require between 5 and 30 minutes to erase an EPROM. Some types of chips take longer to erase than others. An EPROM based part (a PLD or Microcontroller) with a security bit feature is designed so that the security address is typically the*

last bit to be erased. If the window of a chip is not clear, try cleaning the window with alcohol or a solvent. Erase chips if the chips are exposed to sunlight and fluorescent light for months or years; your chips can be erased. You should cover the window of the programmed chips with an opaque label to make the data permanent. Some EPROM based parts can't be erased because they do not have a window. These chips are called one time programmable (OTP) EPROMs.

An EPROM has a quartz window located on the chip just above the die. An EPROM is erased by exposing it to high-frequency ultra-violet light waves. Erasing an EPROM usually takes from 15-20 minutes, but may be shorter or longer, depending on the device. Many manufacturers make EPROM erasers. If you wish to purchase an eraser, call E. E. TOOLS at 408-734-8184. When an EPROM is not being erased, the window may be covered with an opaque label. Sometimes (over a period of years) an EPROM will start to erase due to the level fluorescent light in the room. Direct exposure to sunlight also has this effect and happens much more rapidly and commonly.

NOTE: In order to decide if the device is blank, the user should read the target device. If the buffer is filled with all FFs or 00s, the device is most likely in an erased or unprogrammed state; otherwise, the device is not erased.

CAUTION: Some devices such as Philips P98C52 can pass the BLANK CHECK routine after they are secured even though they are not blank.

Device / Program

Program command will enable you to place new data from the memory buffer into the target device. The BUSY GREEN LED will be blinking during programming.
Make sure the device is correctly inserted into the ZIF socket and the latch is down. Then check the buffer device address range before you start. The values will default to the size of the device.

NOTE: ***<MOTOROLA MICROCONTROLLERS>***

The window of windowed devices must be covered with an opaque label during operation at all times.

NOTE: For all DEVICE/FUNCTION operations, the ERROR YELLOW LED, located at the bottom of the ZIF socket is used to indicate the status of the complete operation. It will turn on if an error has occurred; otherwise it will remain off.

■ **Memory device**

The target device must be blank checked unless the part is electrically erasable. Although most of EEPROMs and Flash Memory devices have the ERASE function in the menu, some EEPROMs such as AT28CXXX or AT29CXXX don't have the ERASE function. Note that EEPROMs without the ERASE function are automatically erased before programming.

■ **Programmable Logic Device operation**

After programming is complete, verification should be performed according to the semiconductor manufacturer's specifications. In order to test vectors, a vector test should be performed (See vector test under the TEST menu). Finally, the part may be secured so that its content can no longer be examined or modified. The security function will not execute if the device fails to verify or pass the vector test properly.

■ **28C256, 28C010, etc.**

28CXXX family devices support Software Data Protection. The user has an option of either protecting or not protecting the data. This option must be changed before the start of any programming operation. To change this option, go to the Option selection under DEVICE/FUNCTION menu and make any changes accordingly. To obtain more information about Software Data Protection, please consult the device manufacturer's specification.

■ **Microchip PIC devices**

Microchip PIC series is different from other Microcontrollers in that they have an EPROM area as well as a CONFIGURATION FUSE. The configuration fuse in the PIC family is used to setup **Oscillator Type, Memory Code Protection, Watchdog Timer, or Processor Mode,** and etc. After

programming the EPROM portion, change the fuses of the items listed under **Option**. Then you must program the configuration option in the **Option** menu.

Perform the following procedure:

1. Program the main memory
2. Click on the OPTION button
3. Set all of the configuration fuse in OPTION menu
4. Click on the Program configuration fuses button

You may also read the status of the Configuration Fuse under the OPTION selection.

In order to obtain more information about programming the configuration fuse, contact Microchip technology at 602-786-7200 or consult the appropriate data book.

Copy from a master chip to a new chip

1. Select the master device from select menu in Microcontroller.
2. Put the chip on the ZIF socket.
3. Click on the **Device** button and read the chip.
4. Click on the **Option** button and read the fuses.
5. Write down all of the option fuses [the memory protect must be disabled] in order to copy the information from your master chip.
6. Place a new chip. It must be the same chip as the master chip.
7. The buffer still holds your master data and **the memory portion.**
8. Click on **Option** again and set all the fuses that you wrote.
   (To change the option, use the arrow button in the selection box.)
9. In the same **Option** menu, Click on the **program configuration fuses**, read and compare the fuses with your original device.

*CAUTION: The **PIC16C711**, will be used as an OTP (one time programmable) chip when you erase the secured device. You cannot reuse the chip after erasing it, even though the PIC16C711 is an erasable device.*

- **Serial EEPROMs**

These devices are electrically erasable, but they operate serially rather than parallelly.

- **Atmel or Xilinx 17xxx**

You need to set the POLARITY FUSE with this family via the Option menu. After programming the main MEMORY, go to the

OPTION menu and make the appropriate change. On OTP (One Time Programmable) devices, the POLARITY FUSE status cannot be reversed once it has been changed. Even on some of the windowed 7xxx family devices (excluding Xilinx 17xxD/L & 17128), the POLARITY FUSE cannot be toggled. Consult the device manufacturer for further instructions on how to handle the Polarity FUSE.

> *CAUTION: Do not touch or remove a device during an operation nor when the BUSY GREEN LED is on.*

Device / Read

Read the data in the source device mounted on the ZIF socket into the buffer for examination.
The checksum will be displayed on the checksum line. The buffer may be edited, saved to a disk, or used to duplicate the chip.

> *CAUTION: Reading the device into the buffer destroys the buffer contents through the specified range. Make sure everything in the buffer that is needed has been saved.*

PLD test vectors are not stored in a logic device; therefore, they cannot be read. The test vector buffer will be empty after reading the PLD.

> *NOTE: Devices that have been secured cannot be read properly. Secured chips may appear all blank, fully programmed, or scrambled.*

Device / Verify

Assures that data in the device matches data in the memory buffer. If your device has the security fuse blown, a verification error is detected. The verify operation requires that the exact data pattern or file that was used to program the device be resident in the memory buffer.

Device / Data Compare

Compares the data in device to the data in buffer and saves any difference into the COMPARE.TXT file. When you have a verify error during the ***Verify*** operation, the ***Data Compare*** command will be useful. It will detect a difference between the

device content and the buffer content and will write the difference into the COMPARE.TXT file under the **ChipMax** directory. You may view the file using an edited utility software.

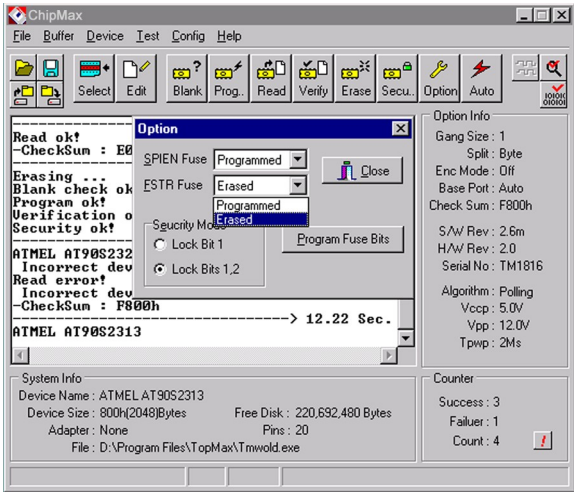
Device / Erase

This option erases the data in your socket before programming it. This operation is valid for only limited devices such as EEPROM, Flash Memory, GAL, PEEL devices. EPROMs that have a window should be erased by UV EPROM erasers externally (see NOTE "Erasing EPROMs" in this manual).

Device / Security

Secures a PLD or Microcontroller so that their content can no longer be examined or modified. Security is confirmed when valid data can no longer be read or verified against a previously read pattern. To ensure that the security fuse has been blown, the ***Security*** operation is preceded by a "read" of the device and followed by a "verify".

> *NOTE: Usually, on a UV erasable PLD or a Microcontroller, a secured device may take longer to erase because the security bit address is designed to erase last.*

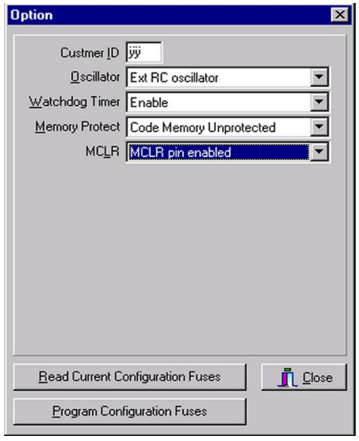


CAUTION: Some devices, such as Philips P98C52, can pass the BLANK CHECK routine after they are secured. Securing a device separates the programmed data pattern from unauthorized access. This command appears only when the selected device supports it. Some Microcontroller's and PLDs can be secured by programming a special address location. The security bit will be cleared when the device is erased. Once a device is secured, it cannot be unsecured to read, verify, or duplicate. Also the secured device is seen as a blank chip even though it is not actually blank.

Device / Encryption

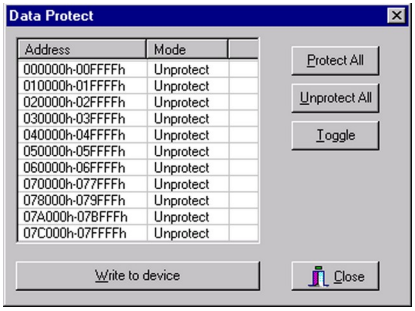
The encryption table is a feature of the 87C51/87C52 family Microcontroller devices. The Encryption array of the Microcontroller is initially unprogrammed (all '1's). In order to protect the code from being easily read by anyone other than the programmer, this feature allows you to program the encryption table that is exclusive NORead with the program code data as it is read out. You have to know its content in order to correctly decode the program code data. Thereafter you will have to use the same displayed encryption array any time you need to read back the device.

Device / Option



Device/Option /Customer ID
The user can store checksum or other code-identification numbers.
Device/Option / Oscillator
Most PIC device family's can be operated in four different oscillator modes. The user can choose one of the following modes from the factory.
LP: Low Power Crystal
XT: Crystal/Resonator
HS: High Speed Crystal/Resonator
RC: Resistor/Capacitor
Device/Option / WATCHDOG TIMER (WDT)
WDT is a configuration bit of special features for PIC device family's.
Device/Option / POWER-UP TIMER
This is a special feature for the PIC device family.
Device/Option / Memory Protect
This configuration fuse can be used to protect against spurious EEPROM writes.

Device/Option / Data Protect

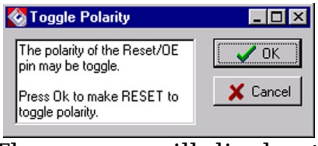


This feature may be enabled or disabled by the user; when shipped from an IC manufacturer, the DATA PROTECT feature is disabled.
Devices have "Software Data Protection (SDP)"
Provides software features to protect nonvolatile data from in advertent writes.

Disable: The SDP command will not protect the entire memory array.
Enable: The SDP command will protect the entire memory array.
Certain Flash device has "Sector Protection"
Protect All: By pressing the Protect All, no data will be erased or written into the device. You can still read data from the device. The block(s) doesn't allow the device to be erased or programmed
Unprotect : The selected block(s) allows the device to be erased or programmed.
Toggle: Change the block status in reverse.
Write Device: The selected block status will be written in the device.
Close: Exit to main menu.

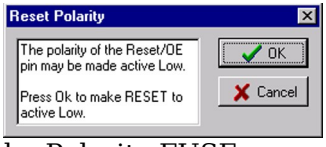
Device/Option / Reset Polarity

Toggle Polarity (for Atmel FPGA): Pressing the button will change the current Polarity Fuse status.



The screen will display the polarity fuse status after changing it.

Reset Polarity (for Xilinx FPGA): The Polarity Fuse is ACTIVE HIGH when shipped from an IC manufacturer. Pressing the button will change the ACTIVE HIGH fuse to ACTIVE LOW fuse. Once it changes to ACTIVE LOW, it may not reset the ACTIVE LOW fuse to ACTIVE HIGH. On OTP (One Time Programmable) devices, the POLARITY FUSE status cannot be reversed once it has been changed. Even on some of the windowed 7xxx family devices (excluding Xilinx 17xxD/L & 17128), the POLARITY FUSE cannot be toggled. Consult the device manufacturer for further instructions on how to handle

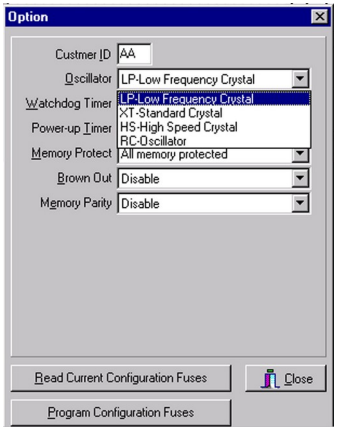


the Polarity FUSE.

Device/Option / Drown Out

This is a special feature for the PIC device family.

Device/Option / MCLR
This is a special feature for the PIC device family.

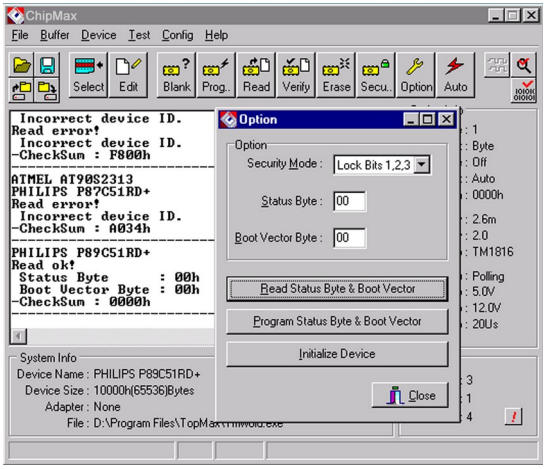


Device/Option / Memory Parity
This is a special feature for the PIC device family.
Device/Option / Low Voltage PGM
This is a special feature for the PIC device family.
Device/Option / FLASH Write Enable
This is a special feature for the PIC device family.
Device/Option / Background DBG
This is a special feature for the PIC device family.
Device/Option / Brownout Voltage
This is a special feature for the PIC device family.

> *NOTE: All the Options above are described in the device manufacturer's data book. Make sure that you understand all configuration features before setting the configuration fuses.*

Option Item/Read Current Configuration Fuses
In order to have the information of all configuration fuses, press this button and you can see all configuration data of the current device. Be sure that you remember all the fuse's information if you want to copy the configuration information

Option Item/Program Current Configuration Fuses
Pressing this button will store all configuration fuse's information in the current device located in the programmer socket.



.

Device / Auto

Auto command will enable you to do the following steps sequentially and it is useful to program a volume quantity device with the same data.
ERASE
BLACK CHECK
PROGRAM
VERIFY
SECURITY

CAUTION: Some devices such as Philips P98C52 can pass the BLANK CHECK routine after they are secured without being blank. Securing a device prevents the programmed data pattern into the device from unauthorized access. This command appears only when the selected device supports it. Some Microcontrollers and PLDs can be secured by programming a special

address location. The security bit will be cleared when the device is erased. Once a device is secured, it cannot be unsecured to read, verify, or duplicate. Also the secured device is seen as a blank chip even though it is not actually blank.

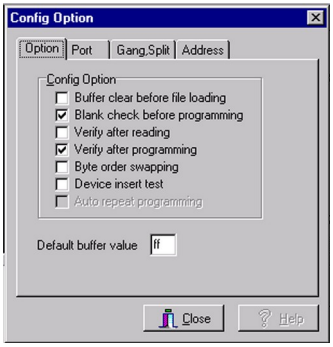
Config

Config Option

Config Option / Default Buffer Value

Fill the buffer value (hexadecimal) with the initial data that you type in this field. This feature helps the user who wants to have different initial values ('00' or 'FF') in the buffer. Once you have made this change, exit to DOS, and restart the ChipMax to get the result of the change.

Config Option / Buffer Clear Before File Loading



When loading a file into the buffer, executing the ENABLE option fills the buffer with the data that is defined in Default Buffer Value before the file is loaded into the buffer. When you load a file that is smaller than the current buffer size, the unfilled buffer will contain the Default Buffer Value so that you may examine the buffer data more conveniently. DISABLE option keeps the same data for the unfilled buffer area after Buffer Load command is executed.

NOTE: Buffer Clear means that the current buffer will be filled with the Default Buffer Value. It can be any data of Hexadecimal values such as FF, 00, or XX

Config Option / Blank Check Before Programming
Enabling Blank Check Before Programming verifies whether the device is erased before programming. Disabling Auto Blank Check Before Programming prevents this check from occurring.

Config Option / Verify After Reading
Setting the configuration menu to ENABLE will allow you to verify whether the device data is the same as the data in your current buffer after reading the source device.

Config Option / verify after programming
Setting the option to ENABLE will allow you to verify whether the device data is the same as the one in your current buffer after programming a device.

Config Option / Byte order swapping
This option applies only to 16-bit wide (E)EPROMs or Flash Memory. User data is displayed in the buffer according to the Intel convention with the default value set at **Disable.**
Enabling this option allows you to use data according to the Motorola convention during ***Program*** and ***Verify*** operations under the ***Device*** selection. However, the data in the buffer is not physically swapped.
When enabled, the MSB (Most Significant Byte) of data is located to EVEN addresses (0,2,4,...) and the LSB(Least Significant Byte) of data is located to ODD addresses(1,3,5,...). For example, Byte swap is useful if an assembler creates a file in Intel format, in which the low byte is read before the high byte, but the file must be in Motorola format, in which the high byte is read before the low byte.

Sample data file (Motorola EXORmacs Format, Code 87):

```
S00B0000441544120492F4FF3
S1130000123456789ABCDEF001122334455667750
S9030000FC
```

Data file opened with format 87 and displayed in the editor (8-bit addressing mode):

```
CURSOR AT LOCATION: 00000000    8 BIT ADDRESSING
                HEXADECIMAL                              ASCII
```

```
ADDRESS   -0 -1  -2  -3  -4  -5  -6  -7  -8  -9  -A  -B  -C -D -E -F
0123456789ABCDEF
 00000000   01 23 45 67 89 AB CD EF 00 11 22 33 44 55 66 77 .#Eg   .   .
"3Duf  w
 00000010   00 00 00 00 00  00  00 00 00 00 00 00 00 00 00 00
```

Example #1: Programming one 16-bit device (Data word width = 16, Odd/even byte **swap = disabled**)

The user data is allocated as follows:

	Device		
	MSB	LSB	
Device Address:	0	23	01
1	67	45	
2	AB	89	
3	EF	CD	

Sample data file (Motorola EXORmacs Format, Code 87):

```
S00B0000444154412049F4FF3
S113000001234567890ABCDEF0011223344556677D0
S9030000FC
```

Data file opened with format 87 and displayed in the editor (8-bit addressing mode):

```
CURSOR AT LOCATION: 00000000    8 BIT ADDRESSING
              HEXADECIMAL                                   ASCII
ADDRESS   -0 -1  -2  -3  -4  -5  -6   -7  -8 -9  -A  -B -C -D -E -F
0123456789ABCDEF
 00000000   01 23 45 67 89 AB CD EF 00 11 22 33 44 55 66 77 .#E.g. . . "3DUf
w
 00000010   00 00 00 00 00 00  00  00 00 00 00 00 00 00 00 00
```

Example #2: Programming one 16-bit device (Data word width = 16, Odd/even byte **swap = Enabled**)

The user data is allocated as follows:

	Device		
	MSB	LSB	
Device Address:	0	01	23
1	45	67	
2	89	AB	
3	CD	EF	

Config Option / Device Insert Test

(under development)

When enabled, this test will allow the ChipMax to first examine the physical position of a device as it is sitting in the programming socket when the user attempts to take any action to that device. Once it has finished examining, the ChipMax will prompt the user for corrective steps if needed depending upon the position of the device. For example, if the device is put in reverse or in the wrong slot of the socket, ChipMax will remind the user to "Insert Device Correctly" or if the number of pins of that device is different than that of the one selected, the user will be prompted with the message "Insert Device." Furthermore, the user will also be prompted to "Insert Device" if the level of the ZIF socket is not latched down. However, if nothing is found wrong with the device, ChipMax will display "Device is found" and proceed with the selected operation.

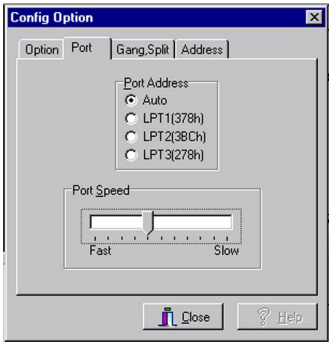
Config Option / Auto Repeat Programming

(under development)

Setting this option to ENABLE allows you to do the same routine continuously by replacing & closing the ZIF socket. This command is useful for production programming. In order to program multiple devices continuously with the same data in the current buffer, set this option ENABLE and program the target devices as AUTO mode in the FUNCTION menu. By replacing the programmed device with a new (erased) device, AllMax software allows you to program the same type of devices without other commands.

> *CAUTION: Auto Repeat Programming will blow the security fuse in GALs and PALs. The* **AUTO** *mode in Device menu execute Blank Check, Program, Verify, Security commands automatically. If you don't want to blow the security fuse of your devices, set this mode to DISABLE (UnClicked) and program your chip manually step by step in* **Device** *menu.*

Config Option / Port



The parallel port address is determined by the ChipMax software.
Auto: ChipMax will select a valid parallel port as the default address in your PC.
LPT1: The parallel port 378 in HEX will be chosen for ChipMax address.
LPT2: The parallel port 3BC in HEX will be chosen for ChipMax address.
LPT3: The parallel port 278 in HEX will be chosen for ChipMax address.
Port Speed: Because the ISA-bus clock speed is not as fast as that of the CPU, we designed this option to facilitate the problem caused when using a fast computer such as Pentium 90/133/166 MHz. The default value is 0. For computers that have CPU speed of greater or equal to 133 MHz, we recommend that you set the Port Delay to 40. In most cases, this option will help to solve the communication problem between your PC and the ChipMax.

> *NOTE: The ChipMax power switch should be ON. The parallel cable is connected between ChipMax and your PC parallel port. Make sure that the shields on each side of the cable are locked. See section 6 Troubleshooting if you are having difficulty with installation and communication.*

Config Option / Gang Split Select

Gang

CM-4G programs multiple E(E)PROMs up to four devices at the same time. In order to program multiple E(E)PROMs, users must use **CM-4G**. The **CM-4G** is especially useful when it is necessary to program many devices with the same data simultaneously. The **CM-4G** is designed for multiple programming and it does not support set programming.

•CM-4G can only be used for most EPROM/Flash devices and cannot be used for GALs, PALS, etc. The software will inform you when you try to select gang mode for an unsupported device.

• Place the CM-4G on top of the ChipMax with all handles pointing in the same direction as the ChipMax socket.

• If the device requires a PLCC adapter contact *E.E. Tools, Inc.*, to obtain the correct adapter.

• From the software menu Device/Gang Select, select 4-Gang Mode. If you receive the "Cannot be selected for 4-gang" message, then it is not possible to program that particular device with the 4-gang adapter.

• Be sure that all devices are of the same brand, size and part number, otherwise CM-4G will not function correctly.

• If any (one) chip is defective, the CM-4G will not Program/Verify/Blank Check any of the devices and you will get an error message. Also note that a device may be defective even though the BLANK CHECK passes.

NOTE: For 4-gang adapters that support only E(E)PROMs upto 32-pin DIP package, or other packages such as PLCC, SOIC with proper adapters.

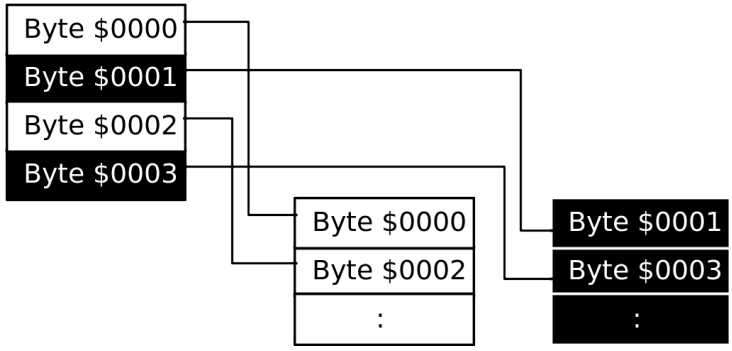
Split

When programming devices for a 16-bit or 32-bit environment, you will need to split your data onto two or four devices.

NOTE: SPLITTING DATA is different from SETTING DATA. Putting buffer data into multiple devices is called "SET DATA". ChipMax doesn't support the SET DATA.

■ Split Programming Examples

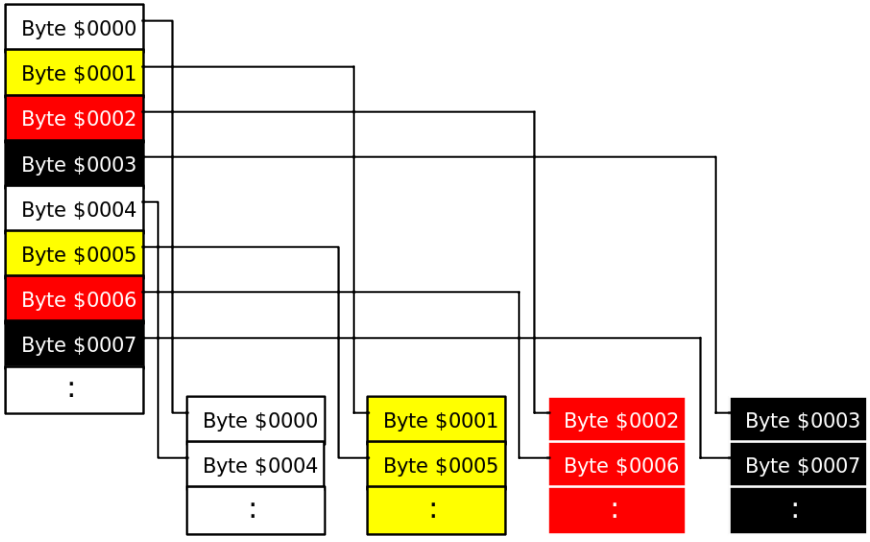
EXAMPLE 1: PROGRAMMING TWO 8-BIT EPROM AS FOLLOWS:



1. Load a 16-bit file into the buffer.
2. Select the target device from menu.
3. Insert the target device (#1) into the ZIF socket.
4. Invoke EVEN in Split data menu.
5. Program the device (#1).
6. Remove the device (#1) and insert the second device (#2) into the ZIF socket.
7. Invoke ODD.
8. Program the second device.

Now, you have two 8-bit EPROMs that have been programmed. The first EPROM (#1) contains all the even address or low bytes and the second (#2) device contains all the odd address or high bytes.

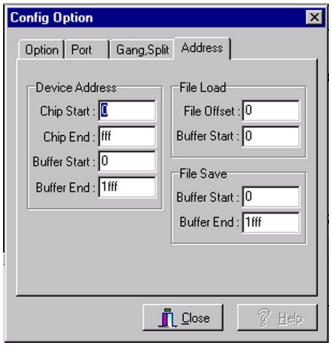
EXAMPLE 2: PROGRAMMING FOUR 8-BIT EPROMS AS FOLLOWS:



1. Select the target EPROM.
2. Load the HEX file (32-bit file) into the buffer.
3. Insert the first EPROM (#1) into the socket.
4. Invoke Word 0 in Split Data menu.
5. Program the mounted device.
6. Remove the programmed device (#1) and insert the second device (#2) into the socket.
7. Follow the same steps as above.

After programming the 4th EPROM with Word 3, you will have four 8-bit programmed EPROMs. The original file (32-bit) is split into four EPROMs that contain 8-bit data in each device.

Config Option / Address



Chip Start: Device Starting address for the data to be programmed in buffer.

Chip End: Device Ending address for the data to be programmed in buffer.

Buffer Start: Buffer Starting address for the data to be programmed.

Buffer End: Buffer Ending address for the data to be programmed.

NOTE: Device size for different devices

Device	Device Address
2716	0 - 7FF
2732	0 - FFF
2764	0 - 1FFF
27128	0 - 3FFF
27256	0 - FFFF
27010/1024	0 - 1FFFF
27020/2048	0 - 3FFFF
27040/4096	0 – 7FFFF

File Load

File Offset is subtracted from addresses from the file downloaded to the programmer. For example, if you set File Offset to 1000h, then the downloaded data minus 1000h would

be placed into the buffer at the address specified by the **Buffer Start Address.**

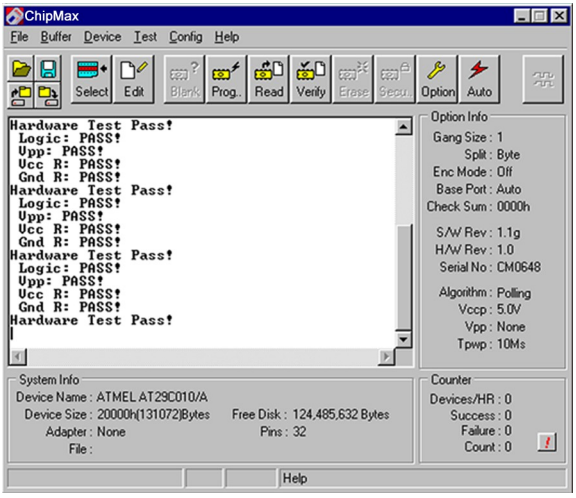
Buffer Start Address is the address in the buffer where you want your downloaded data to start. For example, if you set Buffer Start Address to 800h, then the downloaded data only appears in the buffer beginning at address 800h.

File Save

Buffer Start: Starting address for data to be saved.

Buffer End: Ending address for data to be saved.

Config / Hardware test



A hardware test is designed to assist customers in confirming and diagnosing problems relating to the ChipMax programmer. If a hardware defect with the ChipMax is suspected, the user is recommended to run this test in order to confirm whether or not a problem has occurred with the programmer. The test that pertains to users is the "Pin driver self test." When executed, ChipMax will show a "Hardware Test Result" box in which the status of Pin Drivers (Logic, V12, V34, GND) for each pin from 1 to 40 of the programming socket is displayed. A smiling face symbol (☺) at a particular pin and a particular Pin Driver indicates that the Pin Driver for that pin is "good." An exclamation mark (!) indicates that the Pin Driver for that

pin is "bad." In any event, the ChipMax displays the overall result of the test as either "FAIL" or "PASS" at the bottom of the displayed box.

7. TROUBLE SHOOTING & TECHNICAL SUPPORT

This section provides customer support information such as the return material authorization policy as well as methods of obtaining *E.E. Tools*' technical assistance and software updates. The ChipMax is designed to require a minimum of technical support for both hardware and software. Since we make the product in the USA, we are supplying qualified programmers as trouble-free as possible.

1. Registration

A registration card is located in the user guide manual with the software diskette. Complete the card and return it to *E.E. Tools* to become eligible for:

• Customer support, warranty service and technical assistance
• Notification and special pricing on new products and upgrades

Registration is particularly important if the programmer was purchased from a dealer, a distributor or through your purchasing department. Why not take a moment right now to complete the card?

2. Software Updates

Your programmer is designed to be highly flexible and programmable, allowing it to program a wide variety of chips. Consequently, when a problem does arise, it can usually be fixed with just a free software update. The new software updates are available from our WEB page at **www.eetools.com**, file name: CHIPMAX.EXE or CHIPMAXW.EXE for WIN95/98/NT/2000 version.
Use the new software if you have any other incorrect programming results.

3. Testing the Hardware

Make sure that your programmer works properly before you call us for technical assistance. Refer to Hardware Test section in the Config menu.

4. Quick Self-Diagnostics

In order to provide accurate and fast technical assistance, we recommend that you check the following information before you call our technical support department. We recommend that you obtain the latest software revision before calling our support line with a software problem. Eighty percent of our technical support calls result in asking the user to obtain the latest version of the software.

- Be sure the device selected matches the device being used.
- For a device that uses an adapter, be sure that the adapter is correctly oriented, seated properly, and the ZIF socket lever is down.
- Be sure power cord is securely attached to programmer and power pack to wall socket.
- Be sure that power switch is ON.
- Be sure that the parallel cable (IEEE standard, 25 pins and wires) is correctly and securely attached to the programmer and PC. The connection must be direct; there cannot be any software keys or locks between the parallel port and the programmer. Most switch boxes may also cause difficulties.
- You may need to change your printer port [even though it is working fine with your printer] because the ChipMax communicates with your computer via the printer port in a bi-directional mode.

5. Calling Customer Support

E.E. Tools provides telephone technical assistance during normal business hours (9:00 AM to 5:00 PM, Pacific time).

- Please call our Technical Support Department or your local *E.E. Tools*' distributor while you are at your computer and be prepared to repeat the sequence of steps leading up to the problem

- Have the following information ready when you call:

  - The invoice number for the user who bought the ChipMax from *E.E. Tools*.
  - The distributor's name and the purchased date.

- The model & serial number found in the posterior side of the programming module.
- Your hardware software revision number from the option info screen.
- Description of problem with error message.
- The exact part number you are working with.

6. Service Information

Before sending a unit in for service, call us at 408-734-8184 to obtain a Return Authorization Number (RMA). We will not repair your unit unless an RMA was issued.

Warranty Service: Please return the product in the original package with proof of purchase to the below address. Clearly state in writing the performance problem and send any leads, connectors and accessories that you are using with the device.
Non-Warranty Service: Return the product in the original packaging to the below address. Clearly state in writing the performance problem and return any leads, connectors and accessories that you are using with the device. Customers not on open account must include payment in the form of a money order or credit card. For the most current repair charges contact the factory before shipping the product.

Return all merchandise to *E.E. Tools, Inc.*, with pre-paid shipping.
The flat-fee repair charge includes return ground shipping to addresses in North America only. For overnight shipments and non-North America shipping fees contact E.E. Tools, Inc.

550 Weddell Dr. Suite 5
Sunnyvale, CA 94089, USA

TEL: (408) 734-8184
FAX: (408) 734-8185
Email: support@eetools.com

Include with the instrument your complete return shipping address, contact name, phone number, and description of problem.

7. Limited One-Year Warranty

E.E. Tools, Inc., warrants to the original purchaser that its product and the component parts thereof, will be free from defects in workmanship and materials for a period of one year from the date of purchase.

E.E. Tools, Inc., will, without charge, repair or replace, at its option, defective products or component parts. Returned products must be accompanied by proof of the purchase date in the form of a sales receipt.

To obtain warranty coverage in the U.S.A., this product must be registered by completing and mailing the enclosed warranty card to:
E.E. Tools, Inc., 550 Weddell Dr. Suite 5
Sunnyvale, CA 94089, USA within fifteen (15) days from proof of purchase.

Exclusions: This warranty does not apply in the event of misuse or abuse of the product or as a result of unauthorized alterations or repairs. It is void if the serial number is alternated, defeated, or removed.

E.E. Tools, Inc., shall not be liable for any consequential damages, including without limitations to damages resulting from loss of use. Some states do not allow limitation of

incidental or consequential damages, so the above limitation or exclusion may not apply to you.

This warranty gives you specific rights and you may have other rights, which vary from state-to-state.

Model Number:_____ Date Purchased:_____

8. Useful Web Site Addresses/ Phone Numbers

E.E. Tools.....www.eetools.com
ALTERA.....www.altera.com
AMD.....www.amd.com
INTEL..... www.intel.com
ATMEL.....www.atmel.com
CYPRESS.....www.cypress.com
DALLAS SEMI.....www.dalsemi.com
HITACHI.....www.halsp.hitachi.com
INTEL.....www.intel.com
ISSI.....www.issi.com
LATTICE SEMI. .....www.lattice.com
MITSUBISHI.....www.mitsubishi.com
MICROCHIP.....www.microchip.com
MOTOROLA.....www.motorola.com
NATIONAL SEMI.....www.national.com
NEC.....www.nec.com
OKI SEMI.....www.okisemiconductor.com
PHILIPS SEMI.....www.semiconductors.philips.com
ROHM.....www.rohm.com
SEEQ.....www.seeq.com
SILICON STORAGE..www.ssti.com
ST MICRO.....www.st.com
TEMIC.....www.temic.com
T.I.....www.ti.com
TOSHIBA.....www.toshiba.com
WAFERSCALE.....www.waferscale.com

WINBOND.....www.winbond.com
XICOR.....www.xicor.com
XILINX.....www.xilinx.com
ZILOG.....www.zilog.com

9. Programming Adapter Manufacturers

Compass (Asia).....www.compass21.com
E.E. Tools.....www.eetools.com
Emulation Technology www.1800adapter.com
Emulation Solutions. www.adapters.com
Logical System.....www.logicalsys
Iron Wood.....www.ironwoodelectronics.com

10. EPROM Emulator Manufacturers

E.E. Tools.....www.eetools.com
Tech Tools.....www.tech-tools.com

8. GLOSSARY

BGA Ball Grid Array. A surface mount device with solder balls and a high pin count, similar to PGA.
Bipolar PROM A fuse-link programmable PROM.
Blank Check A test performed by a device programmer to ascertain whether a device has been programmed (partial or total) or is in a virgin state.
Buffer Data storage unit directly stored on CPU.
Checksum A number that results by adding up every element of a pattern. Typically either a four or eight digit HEX number, it is a quick way to identify a pattern, since it is very unlikely that two patterns will have the same checksum.
Compare Reading a programmable device and displaying any discrepancies from the desired pattern. Each error is displayed on the screen. This comparison is slower to perform than a verify on the programmer.
Device Microchip or Integrated Circuit chip.
Die The silicon chip that is located within an IC package. It is a small rectangular flat piece of silicon that has been fabricated with many transistors to perform a specific

function. It is glued into a plastic or ceramic package and connected to the external metal interconnect pins of the IC with very small bonding wires. It can be seen through the window of erasable EPROMs.

DIP Dual Inline Package. An IC package with two rows of through-hole pins, usually on 0.1 pitch, 0.3 or 0.6 inches apart.

FPGA Field Programmable Gate Array. A very complex PLD. The FPGA usually has an architecture that comprises a large number of simple logic blocks, a number of input/output pads, and a method to make random connections between the elements.

Functional Test A test that is performed following the programming of a PLD. The test operates the device in its normal operating mode by simulating the inputs and outputs that the part will experience in normal operation. To perform the test, the engineer must supply a set of test vectors that describe normal operation of the device so the device programmer can apply the specified stimulus and verify that the device is operating as designed. It is important to perform a functional test on PLDs because, in many cases, the PLD cannot be fully tested at the factory before programming so a defective PLD may program correctly but fail the functional test. A properly designed functional test will verify that the part meets the design specification, ensuring that the device, the compiler, the programmer, and the engineer have all performed their respective tasks correctly.

Fuse A metal connection within a PLD or memory that may be melted during programming to break the circuit. These links typically carry input signals to logic gates. Burning all the fuses except those that are required in the desired circuit forms the desired circuit configuration. Since the fuses cannot be tested nondestructively, fuse-like programmable devices cannot be 100% tested at the factory and consequently expected programming yields are usually 98-99%.

GAL Generic Array Logic. EEPROM based second generation PAL devices.

Gang Programmer A multiple-socket programmer that requires each device to be placed in a socket before any can be programmed. See Concurrent Programmer.

HEX file A human-readable ASCII file that represents any binary data. Each byte in the binary pattern is represented by two HEX characters (0-9, A-F) so that any of the 256 possible bytes, which include both control and unprintable characters, may be printed. The HEX file may also contain address or checksum information. The pattern represented by the HEX file may be represented by a binary file or any of the HEX file formats – any file format may contain any pattern. The names of the HEX file formats (Intel, Motorola, Tektronix, etc.) indicate who standardized its format and does not indicate anything about the pattern or the device the pattern is intended for.

I/O Input/ Output.

JEDEC Joint Electron Device Engineering Council (pronounced JED'eck). A group organized by the IEEE (Institute of Electrical and Electronics Engineers) that has defined a standard file format for PLDs.

JEDEC file A file conforming to a standard format that specifies the configuration and testing procedure for a PLD. The file is in a human-readable ASCII format and consists of fields that start with a letter and end with an asterisk. Fields specify the pattern to program into the part, whether to secure the device, a set of test vectors to perform a functional test, and checksums to verify the integrity of the file.

LCC Leadless Chip Carrier. A square ceramic package that has no leads; Instead it has metal areas that are surface-mount soldered to the target circuit. This package is usually used only for military and aerospace applications. Available up to 84 pins.

Memory device A Device that contains an array of storage locations. The device has a set of inputs, called address, which specify which location in the array is being accessed. A set of input/output pins produce the stored number (pattern) when the device is read, and accept a new value when the device is written or programmed. Additionally, there are one or more input pins that select the operating move (read, write, standby, etc.). Memory devices may be classified by whether they are volatile or nonvolatile, and whether they may be erased. The memory's organization refers to its word width and the number of words in the device.

Microcontroller A device that contains a central processing unit (CPU), memory, and I/O ports on a single IC. Microcontrollers that contain any form of nonvolatile memory may be programmed on a device programmer. When connected to a power supply and external crystal, many of these devices form a complete microcomputer.

Non-Volatile The Characteristic of a memory that does not lose its contents when its power is removed. Non-volatile memory is useful in microcomputer circuits because it can provide instructions for a CPU as soon as the power is applied, before secondary devices, such as disk, can be accessed. Non-Volatile memory includes ROM, EPROM and EEPROM.

Oscillator A device that produces an alternating output current.

OTP One-time programmable. The characteristic of a memory device that can be programmed once but cannot be erased. When an EPROM is described as OTP, this means that its die is erasable when exposed to ultraviolet light, but because of its package, which is not transparent, it cannot be exposed to light and thus it cannot be erased.

Package The plastic or ceramic that protects an IC die and connects it to the target circuit.

Parallel printer port A standard port on virtually every PC designed for connection to a printer. This port has eight data lines and several control lines. Parallel ports may be either unidirectional or bidirectional. If your computer has a unidirectional port, the programmer will use the status lines to read data back from the programmer. The port allows high-speed communication (many times faster than a serial port). There may be up to three parallel ports in most PCs designated LPT1, LPT2, and LPT3.

PGA Pin Grid Array. A square, through-hold IC package that has pins located on a square grid with 0.1000-inch pitch. It may have up to several hundred pins. Used primarily for military and prototype designs.

PLCC Plastic Leaded Chip Carrier, A square plastic package that has J-shaped leads on four sides. This can be surface mounted or placed in a socket for through-hole use. Available in 20 to 84 pins.

PLD Compiler A software package that allows an engineer to specify the functionality of a PLD through a high-

level language or schematic diagram. The software will convert the design into a JEDEC or other file for the PLD programmer. PLD compilers are available from numerous IC manufacturers and from third parties. The packages from IC manufacturers support only one brand of device and may be free, inexpensive or expensive. The most popular compiler is PALASM (prices under $200, available from AMD sales offices and representatives) which supports most of AMD's line of PMDs with an easy-to-learn high-level language. The compiler that probably offers the highest level of functionality and flexibility is PLDesigner made by MINC. It supports most PLDs and offers a sophisticated input language with full support for state machines and other complex constructs, partitioning designs into several PLDs, and graphical input. Their tools run on PCs and workstations. PLD compilers have simulators that can be used to test the functionality of your design and validate test vectors that you design before programming a device.

PQFP Plastic Quad Flat Pack. See QFP.

QFP Quad Flat Pack. A square IC package that has surface-mount leads coming from four sides. It is used for high-density applications, usually over 100 pins. Lead pitch may be 0.025 inches or smaller.

RAM Random Access Memory. A volatile memory device.

ROM Read Only Memory. A non-volatile memory device that cannot be programmed by the user. It is programmed at the factory through the use of a mask pattern in the final fabrication steps of the die.

Serial Memory An EPROM or EEPROM that is accessed by shifting in addresses and shifting out data one bit at a time. Interfaces are available using one, two or three wires for clock, data in, and data out.

Socket module An interchangeable metal chassis that contains a programming socket.

SOIC Small Outline Integrated Circuit. A surface-mount IC package that has two rows of leads on opposite sides. Commonly found in 8 to 32 pin sizes. Leads are usually 0.050 pitch.

Test vector A set of characters that describe the inputs and outputs of a device during a functional test. There is

one character in the vector for each pin on the device. Numbers represent inputs to be applied to the device (1 for Vih, 0 for Vil). Letters represent the outputs that must be tested (H for Voh, L for Vol, Z for high-impedance). During the test, the part will be powered up and each input will be applied to the device for the first vector. Then, each output will be applied to the device for the first vector. This process will continue for each vector and any errors will be reported.

TQFP Thin Quad Flat Pack. Similar to QFP but with a lower profile and physically smaller in length and width.

TSOP Thin Small Outline Package. A surface-mount package with fine-pitch leads (usually 0.025 inch pitch) on two sides. This package is very low profile and commonly available in a reverse (mirror image) pinout used to simplify circuit board layout. Usually 32 to 44 pins.

UV Erasable The characteristic of an EPROM that allows it to be erased with exposure to short –wave ultra-violet light. This high-energy light can discharge the floating-gate transistor cells that store bits in an EPROM. The most common source of such light is a mercury vapor tube much like an ordinary fluorescent tube, but without the phosphor that turns the UV light emitted by the mercury into visible light. The light from ordinary fluorescent lamps or sunlight generally takes years to erase an EPROM. All UV erasable parts have a quartz windowed ceramic package that allows exposure with UV light.

Verify Reading a programmable device and comparing its contents to the desired pattern for that device. This is a go/no-go test – it does not report what the discrepancies are. See also: compare.

Word width The number of output pins that a memory device has. The most common size for EPROMs is byte wide (8 bits) and "word" wide, or 16 bits. It can also refer to the aggregate width of several memory devices used in a set.

EPROM ERASERS

MODEL	1. ET240AT-110	2. ET140AT-110	3. LA6T-110	4. DATAII-ACT
Dimension(")	8 x 3 x 3	13 x 5 x 3	7 x 3 x 2.5	6 x 2 x 1
Tray Size(")	6 x 4	3 x 2	3 x 2	3 x 2.5
Case	Aluminum	Aluminum	Plastic	Plastic
# of UV Tube	**2**	**1**	**1**	**1**
# of EPROM	**28**	**16**	**9**	**4**
micro Watts/cm²	**15000**	**9600**	**7000**	**4500**
TIMER	60 min	60 min	40 min	8 min
FOREIGN	ET240AT-220	ET140AT-220	LA6T-220	N/A

TopMax Universal Programmer and Programming Modules

2. Universal Single socket programs all devices(48 pin ZIF):
   All DIP devices up to 48 pins can be programmed directly.
   Different adapters can be inserted into the 48 pin ZIF socket.
3. LED (ON) indicates an operation is done successfully
4. Adapter HEAD contains different sockets (two sockets on one HEAD):
   Usually Four HEADs make 8-sockets. You can find the correct HEAD number from the device selection menu in TopMax software.
5. 8-Gang Socket Module: It programs only EPROM/Flash memory DIP devices up to 32-pin
6. Round Handle
7. Thumb Screw
8. Female Connector

9. 8-Gang Base Module: It holds all different HEADs. It can be replaced with the basic 48ZIF single socket TM-A48.
10. LED (ON) indicates programmer is functioning
11. LED (ON) indicates AC power is on for programmer